Perceptually Motivated Image Features Using Contours

Perceptually Motivated Image Features Using Contours

Xinghui Dong, and Mike J. Chantler

***Abstract*—Dong *et al.* examined the ability of 51 computational feature sets to estimate *human* perceptual texture similarity, however, none performed well for this task. While it is well-known that the human visual system is extremely adept at exploiting longer-range aperiodic (and periodic) "contour" characteristics in images, none of the investigated feature sets exploit higher order statistics (HOS) over larger image regions (>19×19 pixels). We therefore hypothesise that long-range HOS, in the form of contour data, are useful for perceptual texture similarity estimation.**

We present the results of a psychophysical experiment that shows that contour data are more important, than local image patches, or global 2nd-order data, to human observers for this task.

Inspired by this finding, we propose a set of perceptually motivated image features (PMIF) that encode the long-range HOS computed from spatial and angular distributions of contour segments. We use two perceptual texture similarity estimation tasks to compare PMIF against the 51 feature sets referred to above and four commonly used contour representations. This new feature set is also examined in the context of two additional tasks: sketch-based image retrieval and natural scene recognition. The results show that the proposed feature set performs better, or at least comparably to, all the other feature sets. We attribute this promising performance to the fact that the proposed feature set exploits both short-range and long-range HOS.

***Index Terms*—Contours, HOS, image features, perceptual similarity, texture similarity, retrieval, recognition.**

I. Introduction

HIGHER resolution texture similarity estimation seeks to estimate the *degree* to which pairs of textures appear similar to human observers. The performance of common texture features for this particular task does not compare well with that obtained for tasks such as texture segmentation [37] and classification [41], [54] for which they were typically designed. However, the task of perceptual similarity estimation is important and can be used in a number of applications, from measuring the perceived difference between the appearances of textures to simply ranking the results of search engines. Recently, Dong and Chantler [17] assessed 51 computational feature sets and found that the highest agreement rate with human data (obtained using free-grouping experiments [10]) was not greater than 61%, and that coincidently, none of the feature sets tested, exploited longer-range higher order statistics (i.e. HOS computed over areas >19×19 pixels).

It is well known that visual texture can be described using spatial statistics, however, despite over forty years' of research, there is still little agreement as to the type, order or spatial extent over which these statistics should be calculated. First order statistics are computed without reference to the spatial arrangement of pixels and so are rarely used in texture analysis. Second order statistics such as those calculated using the autocorrelation function exploit information concerning periodicities and are often obtained by applying nonlinear functions (variance estimators) to bandpass (linear) filters [36]. These statistics can be computed easily over wide spatial extent. However, higher order statistics are often computationally expensive to acquire, and are thus normally computed within limited spatial extent[1]. Hence "textons" and other vector quantisation methods are typically limited to 19×19 pixel neighbourhoods [17], [19].

Two types of data are therefore commonly utilised for computing texture features: the first comprises 2nd-order data calculated at different scales, while the second involves the estimation of shorter-range aperiodic information. We have found few texture feature sets that capture long-range (>19×19 pixels), aperiodic texture characteristics [17], [19]. It is well-known, however, that these characteristics have an important role in human visual perception [15], [25], [42], [44], [53], [55]. For example, human observers often cannot recognise an aperiodic image when its phase spectrum is scrambled and its power spectrum is kept intact [42], but they are able to exploit the long-range visual interactions evident in contour information [25], [44], [46-47], [53]. Design of perceptually inspired computer algorithms has been studied in the community [2] but, to our knowledge, no research has been reported which utilises contour data[2] for texture analysis.

We therefore hypothesise that "contour" data is important to perceptual texture analysis and we examine this conjecture using two methods. First, we conduct a psychophysical

This work was supported by the Life Sciences Interface theme of Heriot-Watt University. (Corresponding author: Xinghui Dong).

X. Dong completed this work when he worked with the Texture Lab, School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh, UK. He is now with the Centre for Imaging Sciences, University of Manchester, Manchester, M13 9PT, UK (e-mails: xinghui.dong@manchester.ac.uk, and dongxinghui@gmail.com).

M. J. Chantler is with the Texture Lab, School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh, EH14 4AS, UK (e-mail: M.J.Chantler@hw.ac.uk).

[1] However, pyramid decompositions [38] can be utilised to enhance the spatial extent that computational features exploit at the cost of blurring the data used at the higher levels in the pyramid.

[2] In this paper, contour data means the contours extracted from images rather than the gradient magnitude and/or gradient orientation data.

experiment to determine which of three different types of data (global 2nd-order statistics, local higher order statistics and contour data) are more important for human perception. Second, we develop and test a novel feature set that exploits long-range HOS by encoding spatial and angular distributions of contour segments. We assess this feature set's performance against (1) the 51 feature sets that Dong *et al.* [17], [19] examined and (2) four additional feature sets based on shape representation. We also test the feature set in the context of two other popular tasks: sketch-based image retrieval [21] and natural scene recognition [6].

A. Related Work

1) Perceptual Texture Similarity Estimation

Dong *et al*. [17], [19] introduced two evaluation methods for assessing the ability of computational features to estimate perceptual texture similarity: a pair-of-pairs comparison and a texture retrieval task. The two methods used two sets of human-derived data. The resolutions of both datasets were greater than that of the binary similarity data (Same/Different Class) commonly used in either texture classification or retrieval assessments [19]. They tested 51 feature sets but found that none performed well when compared to human-derived perceptual similarity. Their analysis also showed that of the 51 feature sets assessed none used higher order statistics (HOS) derived from local neighbourhoods larger than 19×19 pixels [17], [19]. Additionally, they performed two psychophysical experiments that showed that these types of long-range interactions provide humans with important cues for the perception of texture similarity.

2) Human Perception of Object Outlines

The identification of objects based on outlines has been well studied [15], [55]. Panis *et al.* [44] for instance used outlines to investigate whether or not curved contour segments are important to shape perception. It was found that fragments located at salient points did not necessarily yield better identification performance compared with using fragments placed equidistantly. In addition, sketches, containing outlines, have commonly been used as image retrieval queries, e.g. [21]. Many of these studies above have shown the importance of outlines and sketches to human perception.

3) Computational Shape Recognition Approaches

Contour representation approaches can be divided into two classes: structural and global [57]. Structural contour representation approaches divide a contour into a set of segments that are normally referred to as primitives [5], [9], [32], [35], [39]. In comparison, global methods derive feature vectors directly from whole contours [3], [4], [13], [49], [52]. However, the discriminatory power of point-based shape representation methods [13], [32], [35], [49], [52] is normally affected by noise sensitivity. Furthermore, these approaches have been largely developed to encode individual contours [5], [9], [13], [35], [39]. Hence, none are designed to compute the spatial distribution of the large numbers of contours typically found in textures in a computationally efficient way.

To summarise, none of the shape recognition approaches above can be directly used to represent the spatial layout of the dense contour maps typically found in textures.

Note that although recent deep learning based methods [33], [51] have shown outstanding performance for many computer vision tasks, we do not consider these here, as (1) we are primarily interested in developing feature sets, and (2) higher resolution perceptual texture similarity data are expensive to acquire, which limits the amount of the training data that it is practical to acquire.

B. Contributions

This study investigates the importance of three different types of image property for the human perception of texture and develops a set of image features based on the most important property. In comparison with conference paper [18], this paper (1) describes the psychophysical experiment that investigates the importance of the three image properties in much more detail; (2) revises the original algorithm by both reducing the feature dimensionality and incorporating shorter range contour characteristics; and (3) generalises the assessment by incorporating two additional popular use cases. The main contributions can be identified as: (1) the confirmation of the importance of contour maps to the human perception of texture, compared with either local image patches or global 2nd-order data, and (2) the development of a set of new perceptually motivated image features which exploit longer-range HOS.

C. Overview

We describe the psychophysical experiment and report the results in Section II. In Section III we introduce the new feature set and in Section IV we assess this feature set using two perceptual texture similarity estimation tasks. We test the generality of the proposed feature set using two additional tasks and report the results in Section V. Finally, we draw our conclusions in Section VI.

II. The Importance of Three Types of Data to Texture Perception

The image properties that texture features commonly exploit normally fall into two categories. The first comprises the type of 2nd-order statistics encoded in power spectra. They are often used by filter-based features [11], [17], [19] and encode both long-range and short-range periodicities. The second category concerns the HOS available from local image patches, i.e. those used in vector quantisation or alphabet approaches, e.g. textons [54] or other local neighbourhood based features [1], [37], [40]. They exploit short-range, aperiodic (and periodic) spatial relationships. However, it has been shown that long-range, aperiodic image characteristics, such as contours, are critical to human perception of imagery [15], [25], [42], [44], [53], [55]. The key hypothesis of this paper is therefore, that contours are important to the human perception of texture and that, in particular, they are more important than the two other types of image property described above. It should be noted that we do not consider phase spectra because the application of phase unwrapping to this type of task is an open problem [56].

We therefore used three sets of stimuli (property images) in our experiment with human observers. Samples of each are shown in Fig. 1. Set 1 are phase-randomised (power-only) images, that is they only contain 2nd-order statistics [42] but no HOS. Set 2 comprises randomised, blocked images [17]. These images are divided into blocks which are randomly shuffled. They are therefore unlikely to contain any of the longer-range interactions evident in the original images, but do contain short-range 1st-order statistics, 2nd-order statistics and HOS. Set 3 consists of contour maps of the original images which emphasise longer-range HOS interactions, should they exist.

In order to determine which image property is most important to texture perception, a two-alternative forced choice (2AFC) experiment was conducted. In each trial the observer was shown a quarter image of an original texture, along with a non-overlapping quarter image derived from Sets 1-3. The task of the observer was to decide whether the quarter derived from Sets 1-3 represented the original texture or not. These stimuli and procedure are described in greater detail below.

A. Experimental Design

1) Stimuli

We used the *Pertex* database [29] of 334 textures, as it provides texture images together with higher resolution similarity data derived from a human grouping exercise [10].

Phase-Randomised Images These images were derived using the method introduced by Oppenheim and Lim [42].

Randomised Blocked Images These images were generated by first blocking the image with a green grid and then randomising the position of the blocks in the grid [17]. The reasons for using green rather than the other psychological primary colours are that (1) it is more comfortable on the eye and impairs human perception less; and (2) it makes the grid easy to distinguish from the grey texture. The thickness of the grid was set as three pixels. In addition, the size of the block was set to 19×19 pixels which is the largest neighbourhood exploited by the 51 feature sets (excluding filtering-based features) examined by Dong *et al*. [17], [19].

Contour Maps The Canny edge detector [7] was used to extract edge information from the *Pertex* textures. These edge data were in turn used to construct individual contours (see Section III-A) and the contours aggregated to provide what we refer to in this paper as a "contour map".

2) Procedure

The experiment was divided into three sessions. Phase-randomised images, contour maps and randomised blocked images were utilised in the three sessions in turn. In each session, an observer conducted 334 trials. In each trial, the observer was required to compare one original texture image and its, or another texture's, property image and decide whether or not the property image represented the original. A 2AFC experimental design was employed. If the observer chose "yes", they pressed the left key "←"; otherwise, they pressed the right key "→". The system exited after all 334 trials were performed.

3) Reducing Bias

We used three processes to reduce bias. (1) For a randomised selection of half of all trials we presented an original texture image next to one of its property images, while for the other half of the trials the property image and the original image were derived from different textures. (2) The ten observers were divided into two equal-sized teams. The sequence of the use of the trials was reversed in the second team. In addition, each of the teams performed the trails in three sessions that were conducted at an interval of no less than seven days. With the help of these strategies, the learning effect was reduced. (3) Each original or property image was divided into four equal-sized 512×512 quarters. Throughout the three sessions, the top-left quarters of original images and the bottom-right quarters of property images were employed in order to avoid, or at least inhibit, observers from comparing the original and property images pixel-by-pixel.

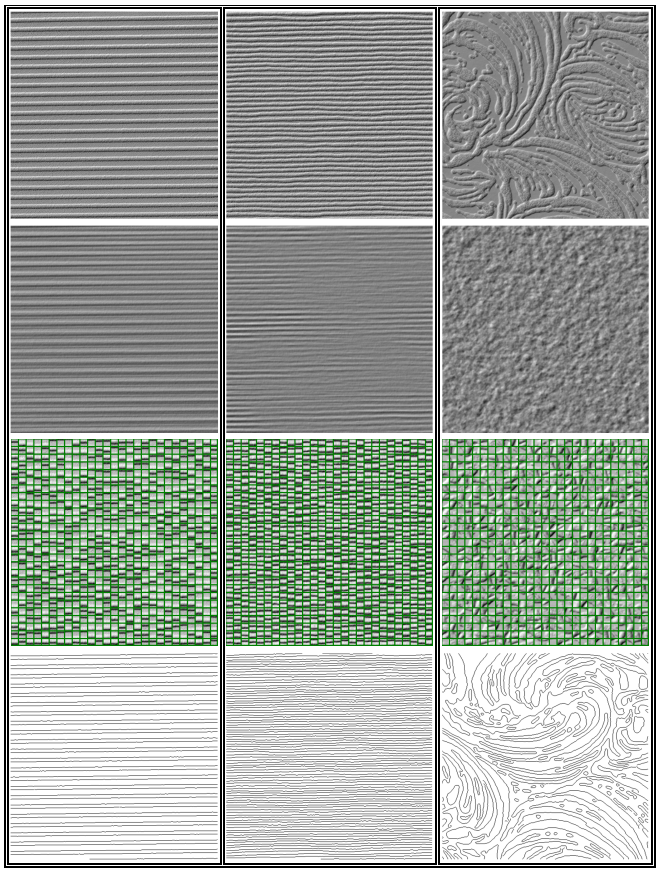


Fig. 1. Each of the three columns shows four images derived from the same texture (although not the same physical texture area). Each of the four rows shows the original image, phase-randomised image, randomised blocked image and contour map in turn.

4) Observers

Throughout the three sessions of this experiment, ten PhD students with normal or corrected-to-normal vision were used. All ten observers signed a consent form before they started the experiment. Each observer was given a 15 GBP Amazon voucher after they completed the experiment.

5) Experimental Setup

Equipment All stimuli were displayed on a calibrated NEC LCD2090UXi monitor at a resolution of 512×512 pixels. The monitor has a resolution of 1600×1200 pixels and pixel

dimensions are 0.255mm×0.255mm (i.e. 100 dpi). Thus, all stimuli were 130.56mm×130.56mm when displayed on the monitor. In addition, the monitor was linearly calibrated to unity gamma, using a Gretag-MacBeth Eye-One, with a maximum luminance of 120cd/m^2. In this case the stimulus images appear as if they are lit by lighting conditions similar to those obtained in a bright room.

Environment The distance between the monitor and the observers was set to approximately 50cm, providing an angular resolution of around 17 cycles per degree. Thus, the stimulus images subtended an angle of 14.89° in the vertical direction. The eyes of the observers were located approximately along the axis of the centre of the screen. The experiment was carried out in a dark room with opaque, matte, black curtains and matte walls without apparent specular reflections.

B. Experimental Results and Analysis

1) Results

A voting process was used with each texture to score the property types. For each texture, if (1) the original image and property image are derived from the same texture, and (2) at least four out of the team of five observers indicate that the property image represents the original, then we count that texture as being well represented by its property image and the score of that type of image property is incremented by 1; otherwise it is assumed that the texture is not well represented by its property image. The experiment was performed in three sessions, with each session using a different type of property image as described in Section II-A-1. Table I reports the scores for each of the image properties.

TABLE I
THE SCORES OF TEXTURES (FROM 334 PERTEX TEXTURES) THAT CAN BE RECOGNISED USING THREE DIFFERENT TYPES OF PROPERTY IMAGES

Subset	Score
Contour Maps	247
Phase-Randomised Images	207
Randomised Blocked Images	157

2) Analysis

We use "Image Comparison Accuracy" (%) to measure the importance of image properties. We define this as the percentage of the textures that are chosen by the observer as the texture that can be represented by its property image compared with random chance (i.e. 167 textures or half of the 334 textures). Fig. 2 shows the average Image Comparison Accuracies and 95% confidence intervals obtained using the three sets of property images across the ten observers.

A one-way repeated-measures ANOVA (Analysis of Variance) [24] was conducted in order to test the significance of the effect of the image property on the Image Comparison Accuracy. The results of Mauchly's test [24] indicate that the assumption of sphericity was satisfied, $\chi^2(2) = 2.90$, $p > 0.05$. The results of the ANOVA show that the Image Comparison Accuracy was significantly affected by the type of image property, $F(2, 18) = 11.84$, $p < 0.05$. Furthermore, the results of the post hoc tests performed using the Bonferroni correction [24] reveal that the Image Comparison Accuracies obtained using randomised blocked images and phase-randomised images were significantly different from that obtained using contour maps, $p < 0.05$. However, there is no significant difference between randomised blocked and phase-randomised images, $p > 0.05$.

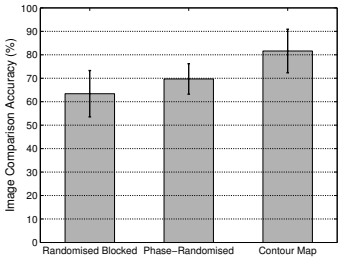


Fig. 2. Means and 95% confidence intervals (error bars) of the Image Comparison Accuracies (%) obtained using three types of property images.

C. Comparison with Perceptual Groups

In order to provide insight as to the ability of the three different types of image property to represent different *types* or *perceptual groups* of texture we break down each subset shown in Table I into 14 subgroups according to the clustering provided by Dong *et al*. [19]. They clustered the 334 *Pertex* textures into 14 perceptual groups by applying a simple hierarchical clustering analysis [26] to the human-derived similarity matrix [10]. We normalise the size of each subgroup using the size of the corresponding perceptual group in order to derive a "Group Image Comparison Accuracy" (%) for the subgroup. These are provided in the bar chart shown in Fig. 3 which also shows representative textures of each perceptual group.

It can be seen that (1) the contour map can represent not only periodic textures (see Cluster 3) but also aperiodic textures (see Clusters 9, 10 and 12); (2) phase-randomised images are generally able to represent periodic and aperiodic but well-ordered textures (Clusters 5 and 6); and (3) the randomised blocked images can represent both periodic and aperiodic textures (see Clusters 6, 8 and 14) but are the least representative type of property image.

The most important point however, is that contour maps provided significantly more relevant information to observers than the other two types of property image, which allowed them to correctly identify 247 out of the 334 textures.

III. PERCEPTUALLY MOTIVATED IMAGE FEATURES

Section II has shown that contour maps are important for human perception of texture. This section therefore introduces a novel set of contour-based image features that are explicitly designed to make use of longer-range HOS as well as other shorter-range data. Essentially, the features are computed by extracting and encoding each contour as a set of related segments. We use these data in three ways as outlined in Fig. 4(d). First we encode the average shape of the contours using joint segment orientation/distance histograms. These provide data on the long-range HOS (of segments). Secondly, we encode the spatial distributions and orientations of the all of the segments within a local window without regard to which

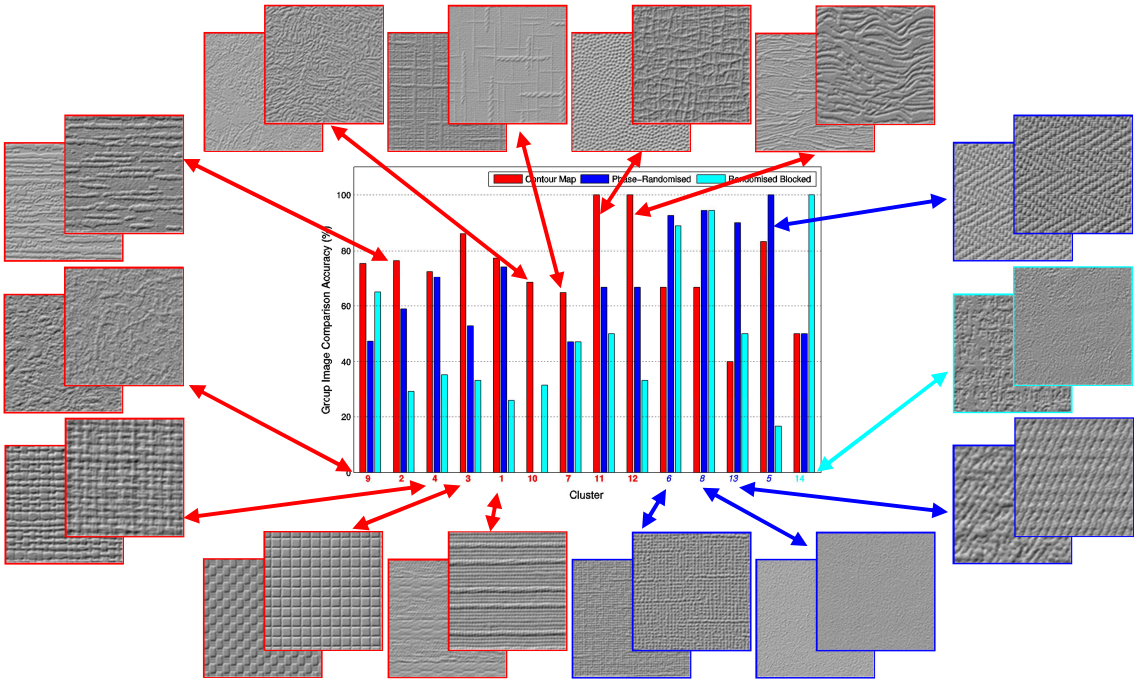


Fig. 3. The bar chart in the centre shows the Group Image Comparison Accuracies for the 14 perceptually grouped clusters introduced in [19]. For each cluster, we show results obtained using contour maps (left and red), phase-randomised images (middle and blue) and randomised blocked images (right and cyan). In addition, we show two representative textures per cluster. These images are outlined using the above colour scheme to indicate which property is most important to their recognition: contour maps (red), 2nd-order statistics contained in phase-randomised images (blue) or the short-range interactions contained in randomised blocked images (cyan).

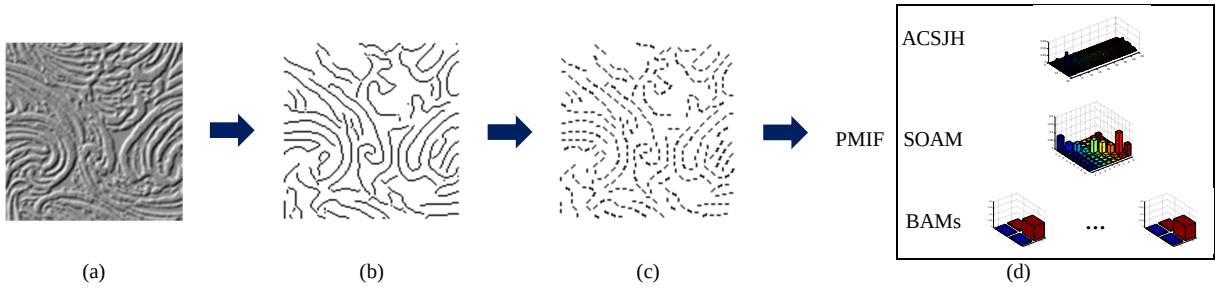


Fig. 4. A representation of the basic information flow: (a) original texture image; (b) contour map; (c) segment map. For display purposes, only a part of pixels are shown for each segment which is approximated by its chord; and (e) three components of the PMIF features: average contour segment joint histogram (ACSJH), segment orientation aura matrix (SOAM) and basic aura matrices (BAMs).

contour they belong. These data provide medium-range (≤23×23 pixels) HOS. Lastly, we encode the spatial distributions of all pixels inside a local window. These data encode short-range, 3×3 pixel, HOS.

A. Obtaining the Contour Maps

The Canny edge detector [7] is utilised to extract contours from a texture image due to its simplicity and effectiveness. A morphological erosion operation [50] is repeatedly applied to the contour map until the output image does not change (Fig. 4(b) shows the result of using a 3×3 neighbourhood). This process removes redundant pixels without allowing contours to break apart.

B. Producing the Segment Maps

1) Traversing a Contour

All contours are traversed from end to end in order to obtain a sequence of contour points as the input of the contour representation method. A number of contours contain branches which make contour representation more difficult. In these cases, all branch points are located and the contours are broken into multiple contours by deleting their branch points.

Connected component labelling [16], with 8-connected neighbourhoods, is performed on the contour map and a connected component is obtained for each continuous contour. The Moore-Neighbour tracing algorithm with Jacob's stopping criteria [28] is applied to each component to provide sequences of points. However, the exterior boundary of one component is derived rather than the component (contour) itself because the tracing algorithm considers each component as a region. The traversing sequence of a contour is obtained from its exterior boundary sequence.

2) Dividing a Contour into Segments

It was found that humans are able to integrate a continuous contour from a series of discontinuous contour segments [25],

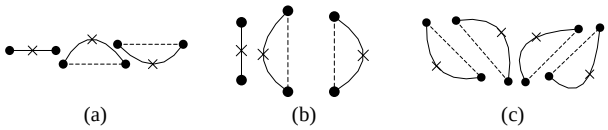
[46-47]. In addition, it has been shown that objects can be identified using discontinuous fragmented contour segments [44]. Thus, non-overlapping segments can retain structure information. Most importantly, representing a set of non-overlapping contour segments is more (computationally) efficient compared with representing a complete set of contour points. For example, the time required for encoding the pairwise spatial relationship (see Section III-C-1) between M contour elements (e.g. points or segments) is in proportion to $(M-1)(M-2)/2$. Given a contour, the M value is smaller when segments rather than when points are used. Thus, using segments is more efficient than using points.

Although primitives or salient points of contours are commonly utilised in their representation [5], [20], [31], the associated computation is relatively expensive and as there had been considerable research using "fragmented" contour segments [57], we decided to use this approach. We first divide a contour into a set of equal-length segments and then encode the spatial distributions and orientations of these segments. Given that a contour contains a sequence of points: $P_1 \ldots P_n$ with coordinates $(x_1, y_1) \ldots (x_n, y_n)$, the length of the contour (CL) is computed as:

$$CL = \sum_{i=1}^{n-1} \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2}. \quad (1)$$

If the length of segments is SL, the contour is then divided into $M = \lfloor CL/SL \rfloor$ segments.

The importance of local orientations to the perception of texture structure has been investigated by Dakin *et al.* [12]. In addition, De Winter and Wagemans [15] found that objects can also be identified using the "straight-line" versions of their outlines. Motivated by these studies, we represented the segments by their mid-point positions (x, y) and chord orientation angles θ ($\theta \in (0^o, 180^o]$). Compared to the chain code method [27], this representation is less sensitive to noise or small variations. Fig. 5 presents three sets of typical segment shapes and their chords. The result is a segment map which encodes each contour as a set of labelled segments, i.e. their mid-point positions and chord orientations. However, as the length of segments increases discriminatory information is lost (see Fig. 6 for example). Hence, only short segments with



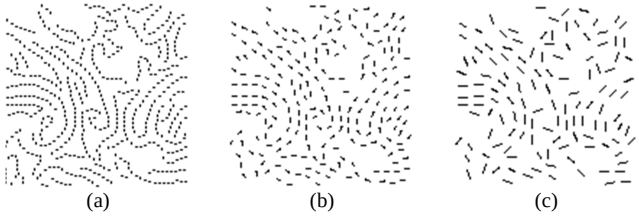
(a) (b) (c)

Fig. 5. Three sets of typical segment shapes and their chords. The solid lines above represent example contour segments, the larger dots represent segment endpoints, the dotted lines show the chords of the segments, while the crosses show the segment mid-points. The orientations of the chords and the positions of the mid-points are used to represent segments.

lengths of 3, 5, 7, 9 and 11 pixels were used.

C. Encoding Contours' Segment Maps

We use three different approaches to represent the spatial distributions and orientations of contours' segments. In the first we compute an average segment distribution across contours (that is we compute pair-wise segment relationships within contours and then average across all contours in an image). In the second we use the aura matrix [22] to compute segment co-occurrence data with no regard as to which contour the segments belong and we restrict the pairs to those occurring within an $L \times L$ local neighbourhood. In the third, we employ the basic aura matrix [48] to represent the spatial distributions of pixels (unit length segments) in a contour map.



(a) (b) (c)

Fig. 6. Three segment maps obtained from the contour map shown in Fig. 4 (b) when the length of segments is set at (a) $SL = 3$ (pixels), (b) $SL = 7$ and (c) $SL = 11$. It is noteworthy that the segments shown are approximated by their chords (only $\lfloor 2 \times SL/3 \rfloor$ central pixels are shown). Each chord is placed at the middle point of its corresponding segment. It can be seen that the ease with which the contours can be identified drops as the length of segments increases.

We refer to these three types of feature using the terms: Average Contour Segment Joint Histogram (ACSJH), Segment Orientation Aura Matrix (SOAM) and Basic Aura Matrices (BAMs). These are defined in the three subsections below, respectively.

1) Encoding the Average Shape of Contours within an Image

We use rate of change of orientation to measure local curvature, and the distance between the mid-points of M segments within a contour is also employed to capture spatial layout. Pair-wise orientation differences and distances are computed for all $(M-1)(M-2)/2$ segment pair combinations. The contour segment joint histogram of the orientation differences and distances is accumulated, and is normalised by the sum of its elements. Note that the orientation angles θ were quantised into A bins and the distances were quantised into B bins, providing histogram resolution of $((2A-1) \times B)$. It is these histograms that are used to represent individual contours. In addition, histograms are averaged across contours to produce a single "Average Contour Segment Joint Histogram" (referred to as ACSJH). See Fig. 4 (d).

2) Representing the Spatial and Angular Distributions of the Segments across Contours

In this feature we compute segment relationships within an image but the mapping of segments to contours is ignored. Since it is computationally expensive to calculate all pair-wise segment data within an image, we instead adapt the Grey Level Aura Matrix (GLAM) as defined in [22] and below, to represent segment-to-segment angle and position relationships.

A GLAM is a 2D (co-occurrence) matrix in which the axes are normally used to represent the two grey levels of pairs of pixels within an $L \times L$ local neighbourhood. The definition of GLAM is based upon the Aura Measure and so we provide both definitions below.

Aura Measure (AM) [22] Given two subsets, $S_1, S_2 \subseteq S$, the AM of S_1 with respect to S_2, is computed as:

$$AM(S_1, S_2, N) = \sum_{S \in S_1} |N_s \cap S_2|, \quad (2)$$

where $|M|$ counts the total number of the elements in M, N_s is

the $L \times L$ neighbourhood at site s and $N = \{N_s, s \in S\}$.

Grey Level Aura Matrix (GLAM) [22] Given that $\{S_i, 0 \leq i \leq G-1\}$ is a set of grey level sets of image $f(s)$, the GLAM of $f(s)$ over N is computed as:

$$GLAM(N) = \left[AM(S_i, S_j, N)\right], \quad (3)$$

where G is the number of grey levels in $f(s)$, $S_i = \{s \in S | f(s) = i\}$ is the pixel set whose grey level is i, and $AM(S_i, S_j, N)$ is the AM between S_i and S_j, $0 \leq i, j \leq G-1$.

In our case we are encoding the joint distributions of the two angles of each pair of segments instead of the grey levels of two pixels. This joint segment angle matrix is accumulated for all pair sets inside a local neighbourhood, where the segment pairs in a pair set are defined by a displacement vector $d = (\Delta x, \Delta y)$ ($|\Delta x|, |\Delta y| \leq \lfloor L/2 \rfloor$). This is similar to the method used for selection of pixel pairs in co-occurrence matrices [30].

We use the term "Segment Orientation Aura Matrix" (SOAM) to refer to the segment angle matrix and its values are used directly in the feature vector. (Note that neighbourhood size was set as $L = 2SL + 1$, where SL was the segment length and $SL \in \{3,5,7,9,11\}$. Therefore the maximum sized neighbourhood considered was 23×23 pixels). Note also that we use the Aura Matrix defined in [22] instead of the Basic Aura Matrix [48] used in [18] and below, in order to reduce the SOAM dimensionality from $A^2 \times (L^2 - 1)$ to A^2.

3) Encoding Spatial Distributions of Pixels in a Contour Map

The features described above, based on the angular distributions of contours' segments, are designed to encode longer range HOS. They are not efficient at capturing information at the micro level. Hence, in order to encode short-range (spatial) interactions between pixels in contours, we use Basic Grey Level Aura Matrices (BGLAM) [48]. They are a special case of GLAM, defined above, but are obtained using a single site neighbourhood system [48]. We use them here because their resolution is higher, they have stronger discriminatory power, and the dimensionality of these matrices is acceptable given that they are computed on the binary valued contour maps. Their dimensionality is $2^2 \times (3^2 - 1) = 32$ (where 2 is the number of grey levels and 3 is the size of local window N_s).

We use the term "Basic Aura Matrices" (BAMs) to refer these 2×2 matrices.

4) Generating the Contour-Based Feature Vector

The ACSJH, SOAM and BAM features are concatenated into a feature vector which we refer to using the term "PMIF" (Perceptually Motivated Image Features). Each PMIF feature vector is normalised by the sum of all elements. In the rest of this paper, "PMIF-A-(SL)" denotes a PMIF feature set in which the segment angle θ is quantised into A bins ($A \in \{9,18,27,36,45\}$) and the segment length, $SL \in \{3,5,7,9,11\}$. It should be noted that PMIF encode long-range, medium-range and short-range HOS.

IV. Perceptual Texture Similarity Estimation Experiments

Three hundred and thirty-four *Pertex* textures [29] and two different similarity tasks [17], [19] were used to assess the performance of the new PMIF feature set against 55 existing feature sets (51 as investigated by Dong *et al.* [17], [19] and four contour type feature sets derived from the shape recognition literature [4], [32]).

These feature sets were used to compute 334×334 similarity matrices, which were used in the two similarity tasks. The first was a pair-of-pairs comparison application [17] and the second was a texture retrieval problem [19]. In the former the classifier is presented with two pairs of textures and must decide on which pair differs most. In the latter, given a query texture, the task is to rank the other textures in the dataset in terms of their similarities with the query texture. One thousand human derived pair-of-pairs judgements [17] and 334 human perceptual texture rankings [19] were used as the ground-truth for the two tasks respectively. Note that it was the availability of these higher-resolution similarity data that dictated our choice of using the *Pertex* texture database [19].

A. The Four Shape Recognition Feature Sets

Shape context [4] and chain code histogram [32] features were calculated for each contour contained in a contour map using both local and global texton dictionaries.

Each contour was represented by a 300 dimensional shape context feature vector or an eight dimensional chain code histogram. The texton generation method proposed by Varma and Zisserman [54] was used to derive ten textons from these features. All 3340 (334×10) textons were concatenated into a texton dictionary. A histogram was accumulated for each contour map using this dictionary. We term the two global texton dictionary based feature sets obtained using the shape context and chain code histogram methods as: "VZ-SC" and "VZ-CCH" respectively. In addition, for the two methods, the ten textons derived from each contour map were used as bins to calculate a histogram from the corresponding features extracted from this map. We refer to the two local texton dictionary feature sets as "SCTH" and "CCHTH" respectively.

B. Computing Similarity Matrices Using Features

Each 1024×1024 texture image was decomposed into five Gaussian pyramid levels using the MatlabPyrTools software package [38]. Each level was separately normalised to an average intensity of zero and standard deviation of one. Feature vectors were computed at all levels and combined into a single multi-resolution feature vector. In addition, the original resolution feature vectors were examined in this study.

The *Chi-square* statistic [54] (see Equation (4)) was utilised to calculate pair-wise distances for histogram-based and the PMIF feature sets, while the *Euclidean* distance (see Equation (5)) was used for all other feature sets. These distances were normalised to $[0, 1]$ and subtracted from 1 to provide data for the similarity matrices. These simple distance (dissimilarity) metrics were used in order not to confound the analysis with the overtraining that might occur with more sophisticated machine learning methods [8] when applied to what is a relatively small texture set.

$$\chi^2(x,y) = \frac{1}{2}\sum_i \frac{(x_i - y_i)^2}{x_i + y_i}. \quad (4)$$

$$Euclidean(x,y) = \sqrt{\sum_i (x_i - y_i)^2}. \quad (5)$$

We test our PMIF features at five different segment angle quantisation schemes (using A bins, $A \in \{9, 18, 27, 36, 45\}$) and five different segment lengths ($SL \in \{3,5,7,9,11\}$). In terms of the five pyramid resolutions, the distances of segment middle points were quantised into $B \in \{80,60,40,35,30\}$ bins respectively. Therefore, the dimensionality of the single resolution PMIF feature set is $(2A-1) \times B + A^2 + 32$.

C. Experimental Design

As the computational similarity matrix is obtained in a different manner from that used to obtain the perceptual similarity data, they are represented in different value spaces. However, direct comparison of the two similarity matrices is avoided when pair-of-pairs comparison [17] or texture retrieval [19] are used, because they use *relative* magnitudes of the similarity data. Having derived the similarity matrices from the computational features it is then a simple task to use these to generate either pair-of-pairs judgements or retrieval rankings.

In the case of the pair-of-pairs comparison the agreement rate [17] between the computational and the human pair-of-pairs judgements was used as the performance metric. For the retrieval based assessment we compared the rankings of the computational and human-derived retrievals (which excluded the query image) using the G measure ($G \in [0,1]$) [23].

$$G = 1 - \frac{\sum_{i=1}^{R}(|r_i - t_i|) + \sum_{i=1}^{N-R}[(N+1) - r_i] + \sum_{i=1}^{N-R}[(N+1) - t_i]}{N(N+1)}, \quad (6)$$

where R is the number of all relevant images in N retrieved images, r_i is the rank order of i-th relevant/irrelevant image retrieved by one search engine (or feature set), and t_i is the "ideal" rank order (i.e. the rank order of i-th texture image ranked by human observers in this research) of the i-th relevant/irrelevant image retrieved. The G measure was averaged over different query textures. We did this for the top $N \in \{10,20,40,60\}$ retrieved textures. The G measure has the advantage that it considers the relative rankings within the two retrievals compared with traditional measures: precision and recall [23] which do not.

D. Experimental Results

1) Pair-of-Pairs Based Evaluation Experiment

Results for two different resolution cases (1024×1024 and the multi-resolution case) are shown in Fig. 7. The two best performing feature sets at these two resolutions, as reported in [17], i.e. Ring and Wedge Filters (RING & WEDGE) [11] and Multi-resolution Simultaneous Autoregressive Model (MRSAR) [37], were used as baselines for our comparison. These results are therefore shown separately in Fig. 7 together with the average performance of the 51 feature sets examined in [17] (as "MeanOf51"). In addition, the results of the four shape recognition-based feature sets (see Section IV-A) are also reported. The remainder of the graph shows the results for our PMIF feature set at five different segment angle bins ($A \in \{9, 18, 27, 36, 45\}$) and five different segment lengths ($SL \in \{3,5,7,9,11\}$).

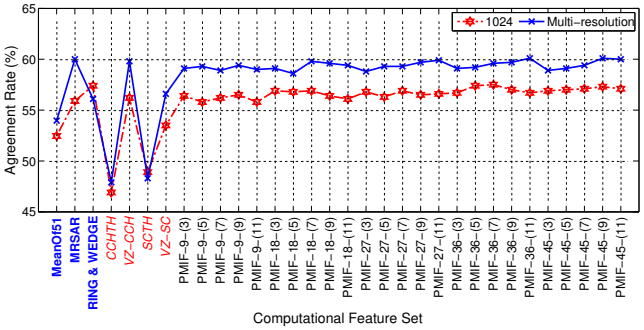


Fig. 7. Agreement rates obtained using computational features against human pair-of-pairs data computed at a resolution of 1024×1024 (red dash-dot trace) and multi-resolution (blue solid trace). The first three columns (labelled in bold blue) show the mean and two best results obtained using the 51 feature sets [17]. The next four columns (labelled in red italic) show results derived using four shape recognition feature sets. The remaining results are derived using our new feature set.

It can be observed that our feature set performs better when the segment angle θ is quantised into 36 or 45 angle bins and when longer segment lengths are used. In these cases it outperforms the two best conventional feature sets. It can also be seen that the performance of all feature sets, excepting RING & WEDGE [11], are enhanced when multi-resolution data are used. In addition, the Varma and Zisserman texton [54] versions of the shape context [4] and chain code histogram [32] termed VZ-SC and VZ-CCH here; perform better than the two local texton feature sets (SCTH and CCHTH). In particular, VZ-CCH closely matches our best feature set in performance.

2) Retrieval Based Evaluation Experiment

In this experiment, the five best feature sets investigated in [19], namely, VZ-NBRHD [54], MRSAR [37], LBPBASIC [40], LBPHF [1] and RING & WEDGE [11], were utilised as baselines. The G measures obtained using the feature sets are shown in Fig. 8 for retrieval sizes of $N \in \{10,20,40,60\}$. From this figure it can be observed that: (1) the use of multi-resolution data improves the performance of all the feature sets; and (2) at 1024×1024 our feature set outperforms all other feature sets with the exception of VZ-NBRHD and

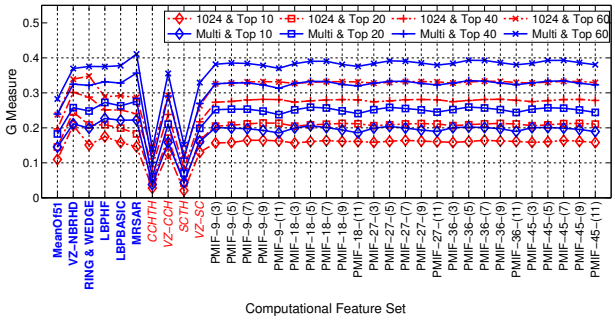


Fig. 8. G measures for the computational features (calculated using human ranking data as ground-truth) provided at a resolution of 1024×1024 (red dash-dot traces) and multi-resolution (blue solid traces). The four different marker types indicate results for four values of $N \in \{10,20,40,60\}$. The first six columns (labelled in bold blue) show the mean and five best results obtained using the 51 feature sets tested in [19] at different conditions. The next four columns (labelled in red italic) show results obtained using four shape recognition feature sets. The remaining results are obtained using our new feature set.

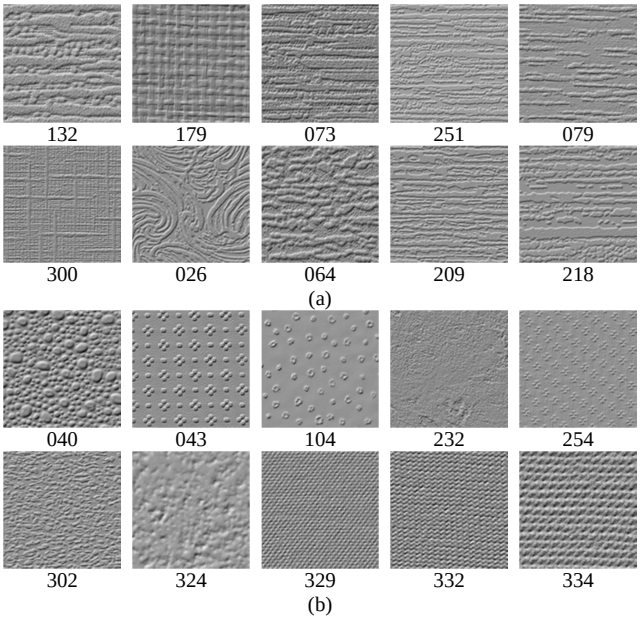


Fig. 9. Best and worst query images (central quarters shown) for PMIF-36-(5) when 10 textures are retrieved: (a) 10 "best" query textures sorted in a descending order of G measures; and (b) 10 "worst" query textures with G measures of 0.

RING & WEDGE feature sets. For the multi-resolution case it outperforms all except the MRSAR feature set.

Fig. 9 shows the top 10 "best" and "worst" query textures for the PMIF-36-(5) feature set. Clearly, the textures with obvious long-range structures (see Fig. 9 (a)) can be retrieved more reliably than those with small blob-like structures ("040", "043", "104", "254" and "302") or without obvious structures ("232" and "324"). In addition, our feature set cannot retrieve those regular textures whose long-range contours cannot be accurately extracted ("329", "332" and "334").

E. Discussion

Here we discuss issues concerning choice of segment length, effect of contour detector and performance measures for PMIF features; the feasibility of incorporating other characteristics into PMIF features; and the merits of the use of contour data. For simplicity, we only consider the pair-of-pairs experiment.

1) Choice of Segment Length

The choice of the segment length is a trade-off between efficiency and accuracy. Short segments require greater computation, while long segments do not allow the original shape of contours to be retained (see Fig. 6). After experimentation, a segment length of five pixels was observed to provide reasonable computational efficiency while allowing flexible encoding (see the experiments described in Section V).

2) Effect of Contour Detection Method

For contour detection, we compared the original Canny detector [7] with (1) a Logarithmic Image Processing (LIP) model based Canny [43] and (2) a Structured Forest (SF) method [14]. All other procedures were kept constant: the segment length was set as $SL = 5$ and the segment angle θ was quantised into $A \in \{9, 18, 27, 36, 45\}$ bins.

The agreement rates obtained are reported in Table II. It can be seen that the SF-based system outperforms the two other implementations while the LIP implementation provides the worst performance. Note that while LIP provides some illumination invariance, the *Pertex* textures [29] were acquired using the identical illumination conditions and this could explain why the LIP version does not have an advantage here.

TABLE II
AGREEMENT RATES (%) OBTAINED USING PMIF WHEN THREE EDGE DETECTORS ARE USED, INCLUDING CANNY, LIP CANNY AND STRUCTURED FORESTS (SF). THE LENGTH OF SEGMENTS IS SET AS FIVE PIXELS WHILE THE SEGMENT ANGLE θ IS QUANTISED INTO $A = 9, 18, 27, 36$ AND 45 BINS.

Edge Detector	$A = 9$	$A = 18$	$A = 27$	$A = 36$	$A = 45$
Canny [7]	59.30	59.30	59.30	59.20	59.10
LIP Canny [43]	58.20	57.70	58.50	58.60	58.50
SF [14]	**59.50**	**60.80**	**60.10**	**60.20**	**60.30**

Bold fonts indicate the highest performance across the three edge detectors.

It should also be noted that the Canny implementations are more efficient than the Structured Forest with the latter implementation being approximately ten times slower. For a machine with a 64-bit, 3.40GHz Intel(R) i7-4770 CPU and 16.0 GB memory, the total time required for extracting contours from 334 *Pertex* textures at five pyramid levels using the Canny, LIP Canny and SF detectors are 210.42, 215.82 and 2195.68 seconds respectively. Thus, the use of Canny vs. Structured Forest can be viewed as another speed vs. accuracy trade-off.

3) Effect of Performance Measures

In the pair-of-pairs experiment, we used "agreement rate" [17] as the performance measure. However, a variety of performance measures could have been used. For example, Spearman's correlation coefficient (CC) [24] is a commonly used measure in vision science. In addition, as a metric of the mutual dependence between two variables, mutual information (MI) is popular in information theory [45].

TABLE III
THE VALUES OF THE AGREEMENT RATE (AR, %), SPEARMAN'S CORRELATION COEFFICIENT (CC) ($\partial = 0.05$) AND MUTUAL INFORMATION (MI) OBTAINED USING SIX BASELINES AND PMIF AT THE MULTI-RESOLUTION SCHEME. ONLY THE BEST PERFORMANCE OF PMIF IS SHOWN IN EACH CASE.

Measure	MRSAR [37]	R&W [11]	CCHTH	VZ-CCH	SCTH	VZ-SC	PMIF
AR (%) [17]	60.00	56.10	47.90	59.80	48.30	56.60	**60.10**
CC [24]	0.3155	0.1970	-0.0704	0.2927	-0.0189	0.1726	**0.3227**
MI [45]	0.0289	0.0106	0.0012	0.0278	0.0008	0.0125	**0.0294**

Bold fonts indicate the highest performance across different feature sets.

We computed Spearman's correlation coefficient and MI using values derived directly from the two sets of similarity data. Table III shows the original pair-of-pairs agreement rate, against the CC and MI values. These were obtained for six baseline feature sets and PMIF. This shows that the PMIF feature set outperforms its counterparts independent of which of the three performance measures is used.

4) Incorporating Other Image Characteristics

The PMIF feature set utilises contour data at the possible cost of distortion of 2nd-order statistics. However, Ojala *et al*. [41] showed that a local variance measure ("VAR") is complementary to the Local Binary Patterns (LBP) feature set (which also removes or distorts 2nd-order statistics). Inspired

TABLE IV
AGREEMENT RATES (AR, %) OBTAINED USING THE PMIF AND PMIF&VAR FEATURE SETS. THE LENGTH OF SEGMENTS (SL) AND THE BINS OF SEGMENT ANGLES (A) ARE SET AS DIFFERENT VALUES.

SL	Feature Set	$A=9$	$A=18$	$A=27$	$A=36$	$A=45$
3	PMIF	59.10	59.10	58.80	59.10	58.90
	PMIF&VAR	**59.40**	**60.10**	**59.90**	**59.80**	**60.10**
5	PMIF	**59.30**	59.30	59.30	59.20	59.10
	PMIF&VAR	59.10	**60.20**	**60.10**	**60.40**	**60.40**
7	PMIF	58.90	**59.80**	59.30	59.60	59.40
	PMIF&VAR	**59.80**	59.50	**60.30**	**60.30**	**60.40**
9	PMIF	59.40	59.60	**59.70**	**59.70**	**60.10**
	PMIF&VAR	**59.80**	**60.00**	59.50	59.50	60.00
11	PMIF	59.00	**59.40**	**59.90**	**60.10**	**60.00**
	PMIF&VAR	**59.20**	59.30	59.80	59.30	59.70

Bold fonts indicate the higher performance between those obtained using the two feature sets.

by this, we added the VAR feature to the PMIF feature vector in order to incorporate local contrast characteristics. The new feature set is termed as "PMIF&VAR". We compared this feature set with original PMIF in the pair-of-pairs experiment. The results are reported in Table IV. It can be seen that the use of the local variance data improves the performance of PMIF when the segment length is less than nine pixels or low numbers of segment angle bins are used. However, this is not the case when longer (≥ 9 pixel) segments and more (≥ 27) segment angle bins are used. This may be attributed to the sparse representation of PMIF when fewer (longer) segments and more segment angles are used.

5) Merits of the Use of Contours

As shown in Section II, the contour data is suitable for representing global image structural information. In this study, we encode each contour using its segments. This approximation obtains computational efficiency but may sacrifice the representation accuracy especially for those small-scale contours (see Fig. 9 (b)). On the other hand, the proposed feature set does not represent an image well when there is no obvious structure in the image (see Fig. 9 (b)). These findings probably explain why the PMIF feature set was slightly outperformed by MRSAR [37] which models local image characteristics based on grey level image patches in the retrieval task. Therefore, a more precise representation of local contour elements and the joint modelling of local image contrast characteristics should improve the performance of the PMIF feature set. However, the time cost for computing PMIF features is lower than that required for MRSAR. For the machine described in Section IV-E-2, the average time cost required for the extraction of PMIF and MRSAR features from 512×512 *Pertex* [29] images are 22.95 and 131.30 seconds respectively. In this context, the PMIF feature set also provides a good trade-off between efficiency and accuracy.

F. Summary

The PMIF feature set performs well in the two experiments when compared to existing feature sets. Although PMIF performs slightly worse than its original version: SDoCS [18], its feature dimensionality has been reduced greatly. This makes the generalisation of it to other applications more practical.

V. GENERALISATIONS

We assessed the performance of the PMIF features in two additional applications: sketch-based image retrieval (SBIR) [21] and natural scene recognition [6]. As the resolution of the images used in these experiments is lower than that used in the previous experiments, the features were only extracted on three Gaussian pyramid [38] levels. The BAM features were extracted using three levels of spatial pyramid [34] (21 sub-images) and were concatenated. All conditions for feature extraction were kept the same except the two aspects above. In this case, the dimensionality of the single resolution PMIF feature set is $(2A-1)\times B + A^2 + 32 \times 21$.

A. Sketch-Based Image Retrieval Experiment

We used the framework proposed by Eitz *et al.* [21] for the SBIR task. In this framework, human ranking data are used as the ground-truth and Kendall's rank correlation coefficients [24] are used as the performance measures. We also employed the G measure [23] used in the texture retrieval task as it is suitable for comparing two non-identical rankings [23]. As in [21], we set $\sigma = 5$ and low and high thresholds for the Canny detector [7] to 0.05 and 0.2 respectively.

First, as baselines, we used the best Kendall's correlation coefficients obtained using five feature sets: the Tensor and HoG (T & HoG) descriptor; the shape context descriptor (SCD); the histogram of oriented gradients descriptor (HoG); the spark descriptor (SD) and the standard histogram of oriented gradients descriptor using dominant local orientations (SHoG) tested by Eitz *et al.* [21] (see Table V). The best results obtained using the shape recognition feature sets introduced in Section IV-A and our PMIF feature set are also reported in

TABLE V
BEST KENDALL'S CORRELATION COEFFICIENTS (τ) BETWEEN COMPUTATIONAL AND PERCEPTUAL RETRIEVALS ($\partial = 0.05$)

Feature Set	T & HoG	SCD	HoG	SD	SHoG
τ	0.223 [21]	0.161 [21]	0.175 [21]	0.217 [21]	**0.277** [21]
Feature Set	SCTH	CCHTH	VZ-SC	VZ-CCH	MRPMIF
τ	0.002	0.012	0.024	0.012	0.231

Bold fonts indicate the highest performance across ten feature sets.

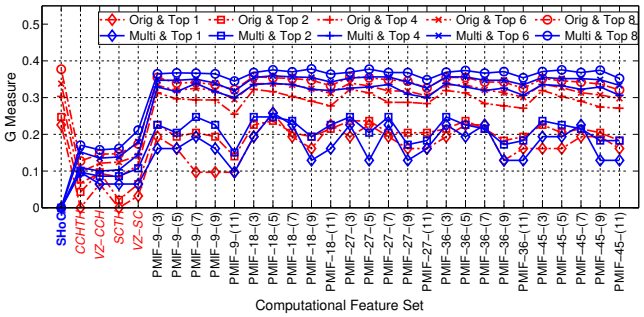


Fig. 10. G measures for the computational features (calculated using human ranking data as ground-truth) provided at the original resolution (red dash-dot traces) and for the multi-resolution (blue solid traces). Five different marker types are used to indicate results for five values of $N \in \{1,2,4,6,8\}$. The first column (labelled in bold blue) shows the best results obtained using SHoG [21] (multi-resolution is not available). The next four columns (labelled in red italic) show results obtained using four shape recognition feature sets. The remaining results are obtained using our new feature set.

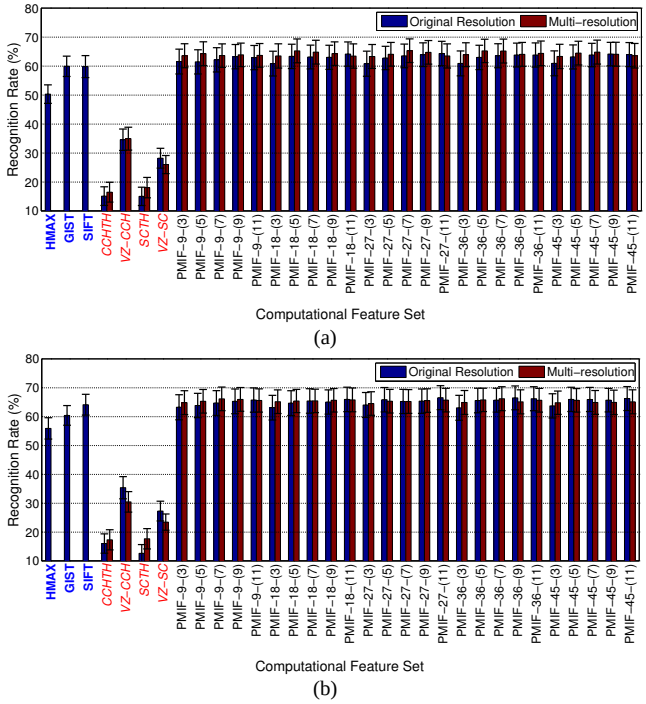
Table V. It can be seen that the multi-resolution PMIF (MRPMIF) outperformed all its counterparts with the exception of SHoG [21] which uses the most dominant sketched lines.

Second, since Eitz *et al.* [21] published the best ranked list obtained using SHoG we compared PMIF and the four shape recognition feature sets with it using the G measure. We examined the top 1, 2, 4, 6 and 8 (out of 40) retrieval images. (The ratio of 8/40 is approximately equal to the 60/334 ratio used in texture retrieval). The results are shown in Fig. 10. It can be seen that the best multi-resolution PMIF outperformed all the other feature sets.

B. Natural Scene Recognition Experiment

Brown and Susstrunk [6] derived a new natural scene image dataset (containing 477 colour and near-infrared (NIR) image pairs) and used this to compare three feature sets: HMAX, GIST, and SIFT for scene recognition. They randomly selected 99 images for testing (11 per category) and trained using the remainder. We used the same experimental scheme but only utilised the nearest-neighbour classifier, and did not use the Bayes or linear SVM classifiers [6]. We conducted the experiment 1000 times, rather than repeating it for only ten times with different training/test splits. The mean and standard deviation of the recognition rates (%) were used as performance measure.

Considering detailed information is necessary for matching two natural scene images, the value of σ of the Canny detector



(a)

(b)

Fig. 11. Average recognition rates (%) and standard deviations obtained using computational features on (a) grey level images and (b) both grey level and near-infrared images. Each bar-group shows two resolutions: original resolution (left), and multi-resolution (right). The first three columns (labelled in bold blue) show the best results obtained using three feature sets tested in [6] (multi-resolution is not available). The next four columns (labelled in red italic) show results obtained using four shape recognition feature sets. The remaining results are obtained using PMIF.

TABLE VI
BEST AVERAGE RECOGNITION RATES (%) AND STANDARD DEVIATIONS

Feature Set		HMAX	GIST	SIFT	PMIF	MRPMIF
RR (%)	L	50.3±3.2 [6]	59.9±3.5 [6]	59.8±3.8 [6]	64.3±4.2	**65.3**±4.1
	LI	55.9±3.7 [6]	60.4±3.4 [6]	64.1±3.6 [6]	**66.5**±4.2	66.2±4.1
	RGB	53.4±3.9 [6]	60.0±3.3 [6]	62.9±3.1 [6]	65.1±4.1	**67.0**±4.2
	RGBI	57.1±4.0 [6]	60.0±4.4 [6]	67.5±2.3 [6]	66.2±4.0	**68.0**±4.2

Bold fonts indicate the best performance for each image data combination.

[7] was set as $\sqrt{2}$. We carried out the experiment using four different combinations of image data: L = luminance (grey level), LI = luminance + NIR, RGB, and RGBI = RGB + NIR [6]. The performances of HMAX, GIST, and SIFT reported in [6] were used as baselines. Fig. 11 reports the results obtained using the three baselines, the four shape recognition feature sets and the PMIF feature sets for the L and LI data. It can be observed that PMIF performed better than all its counterparts on these data. A summary of the results for all four image data combinations is shown in Table VI. This shows the best average recognition rates and their corresponding standard deviations obtained using the three baselines and our PMIF feature sets. Both the PMIF and MRPMIF feature sets outperformed the other feature sets for the four image data combinations.

VI. CONCLUSIONS

In this paper we first examined the importance of three different categories of data to the human perception of texture. Two categories were motivated by the information commonly used by existing texture features: 2nd-order statistics, and short-range higher order statistics (HOS) (typically obtained from image patches). The use of the third category, contour data, was motivated by the fact that the human visual system is extremely adept at exploiting these visual cues [25], [44], [46-47], [53] and that they utilise long-range HOS. We conducted an experiment with human observers that showed that for the *Pertex* database [29], contours are the most useful category of data for human texture discrimination.

Inspired by this result and the fact that none of the 51 feature sets examined by Dong *et al.* [17], [19] use HOS beyond 19×19 pixel neighbourhoods, we developed a set of new image features, based on representing contours as sets of segments. We refer to this feature set as: "Perceptually Motivated Image Features" or "PMIF" for short. The PMIF feature set exploits the long-range, medium-range and short-range HOS available from the segment and pixel distributions.

We tested this feature set using two texture similarity estimation tasks. The first task was a pair-of-pairs comparison in which the classifier simply has to decide which of the two pairs differ most [17]. The second task was image retrieval [19]. Using an existing human-derived higher-resolution similarity matrix [10] we were able to fully rank the results which in turn allowed us to assess the ability of features to estimate perceived similarity more thoroughly. (Note for this comparison we used the "G" measure [23] that takes into account rank order). We also applied the PMIF feature set to two more popular tasks: sketch-based image retrieval (SBIR)

[21] and natural scene recognition [6]. The results showed that the PMIF feature set outperformed, or performed comparably to, its counterparts in the four tasks.

Although the proposed feature set does not utilise dictionary learning [54], parameter estimation [37], or contour selection [21] techniques, it outperformed, or performed comparably to, the existing feature sets examined in this study. The VZ-CCH, MRSAR [37] and SHoG [21] feature sets which performed well in parts of the experiments are relatively computationally intensive, and in these cases, the proposed feature set is more efficient.

While the PMIF feature set was designed for, and is particularly suitable for, the representation of images that contain long-range (aperiodic or periodic) structure; there remain three open problems. First, the feature set cannot encode small contours well, as it uses segments rather than points to describe contour shapes (see Section III-C-1). Secondly, it cannot represent an image well that is devoid of obvious structure (in this case, grey level, or colour information is needed). Finally, longer-range spatial distributions across contours are not exploited within the feature set.

However, we have shown that using image HOS over a range of spatial extent is important both to human perception and for machine analysis, particularly for exploiting the larger scale structures often found in image texture. We hope that this work will encourage further research into the usefulness of such features.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their comments and suggestions to improve this manuscript. The authors would also like to thank Prof. Junyu Dong for his comments and Dr. David Robb for his proofreading.

REFERENCES

[1] T. Ahonen, J. Matas, C. He and M. Pietikainen, "Rotation Invariant Image Description with Local Binary Pattern Histogram Fourier Features," in *Proc. Scandinavian Conference on Image Analysis*, 2009, pp. 61-70.

[2] A. Amiri and S. Haykin, "Improved Sparse Coding under the Influence of Perceptual Attention," *Neural Computation Journal*, vol. 26, no. 2, pp. 377-420, 2014.

[3] K. Arbter, W.E. Snyder, H. Burkhardt, and G. Hirzinger, "Application of affine-invariant Fourier descriptors to recognition of 3-D objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 640-647, 1990.

[4] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509-522, 2002.

[5] S. Berretti, A.D. Bimbo, and P. Pala, "Retrieval by shape similarity with perceptual distance and effective indexing," *IEEE Trans. Multimedia*, vol. 2, no. 4, pp. 225-239, 2000.

[6] M. Brown and S. Susstrunk, "Multi-spectral SIFT for scene category recognition," in *Proc. 2011 IEEE Conference on Computer Vision and Pattern Recognition*, 2011, 177-184.

[7] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679-698, 1986.

[8] G.C. Cawley and N.L.C. Talbot, "Over-fitting in model selection and subsequent selection bias in performance evaluation," *Journal of Machine Learning Research*, vol. 11, pp. 2079-2107, 2010.

[9] N. Chomsky, Syntactic Structures, The Hague/Paris: Mouton, 1957.

[10] A.D.F. Clarke, X. Dong and M. J. Chantler, "Does Free-sorting Provide a Good Estimate of Visual Similarity," in *Proc. the 3rd International Conference on Appearance*, 2012, pp. 17-20.

[11] J.M. Coggins and A.K. Jain, "A Spatial Filtering Approach to Texture Analysis," *Pattern Recognition Letters*, vol. 3, pp. 195-203, 1985.

[12] S.C. Dakin, "Orientation variance as a quantifier of structure in texture," *Spatial Vision*, vol. 12, pp. 1-30, 1999.

[13] E.R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*, New York: Academic Press, pp. 171-191, 1997.

[14] P. Dollar and C. L. Zitnick, "Fast Edge Detection Using Structured Forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1558-1570, 2015.

[15] J. De Winter and J. Wagemans, "The awakening of Attneave's sleeping cat: Identification of everyday objects on the basis of straight-line versions of outlines," *Perception*, vol. 37, no. 2, pp. 245-270, 2008.

[16] M. B. Dillencourt, H. Samet and M. Tamminen, "A general approach to connected-component labeling for arbitrary image representations," *Journal of the ACM*, vol. 39, no. 2, pp. 253-280, 1992.

[17] X. Dong and M. J. Chantler. "The Importance of Long-Range Interactions to Texture Similarity," in *Proc. the 15th International Conference on Computer Analysis of Images and Patterns*, 2013, vol. 8047, pp. 425-432.

[18] X. Dong and M. J. Chantler, "Texture Similarity Estimation Using Contours," in *Proc. 2014 British Machine Vision Conference*, 2014, pp. 49.1-49.11.

[19] X. Dong, T. Methven, and M. J. Chantler, "How Well Do Computational Features Perceptually Rank Textures? A Comparative Evaluation," in *Proc. the ACM 2014 International Conference on Multimedia Retrieval*, 2014, pp. 281-288.

[20] G. Dudek and J.K. Tsotsos, "Shape representation and recognition from multiscale curvature," *Comput. Vision Image Understanding*, vol. 68, no. 2, pp. 170-189, 1997.

[21] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-based image retrieval: benchmark and bag-of-features descriptors," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 11, pp. 1624-1636, 2011.

[22] I M., Elfadel and R.W., Picard, "Gibbs random fields, cooccurrences, and texture modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 1, pp. 24-37, 1994.

[23] R. Fagin, R. Kumar and D. Sivakumar, "Comparing Top K Lists," in *Proc. 14th ACM-SIAM Symposium on Discrete Algorithms*, 2003, pp. 28-36.

[24] A. Field, *Discovering Statistics Using SPSS*, SAGE Publications Ltd, 2009.

[25] D. J. Field, A. Hayes and R. F. Hess, "Contour integration by the human visual system: evidence for a local 'association field'," *Vision Research*, vol. 33, pp. 173-193, 1993.

[26] C. Fraley and A.E. Raferty, "How many clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis?," *The Computer Journal*, vol. 41, no. 8, pp. 578-588, 1998.

[27] H. Freeman, "On the encoding of arbitrary geometric configurations", *IRE Trans. Electron. Comput.*, EC-10, pp. 260-268, 1961.

[28] R. C. Gonzalez and R. E. Woods, *Digital Image processing*, NJ: Prentice Hall Upper Saddle River, 2002.

[29] F. Halley, "*Pertex v1.0*," 2011; http://www.macs.hw.ac.uk/texturelab/resources/databases/pertex/.

[30] R. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification," *IEEE Trans. Systems, Man, Cybernetics*, vol. 3, pp. 610-621, 1973.

[31] X. Hilaire and K. Tombre, "Robust and Accurate Vectorization of Line Drawings", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 6, pp. 890-904, 2006.

[32] J. Iivarinen and A. Visa, "Shape recognition of irregular objects," in *Proc. SPIE 2904*, 1996, pp. 25-32.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems*, pp. 1106-1114, 2012

[34] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognising Natural Scene Categories," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, vol. 2, pp. 2169-2178.

[35] S.Z. Li, "Shape matching based on invariants," *Progress in Neural Networks: Shape Recognition*, vol. 6, pp. 203-228, 1999.

[36] J. Malik and P. Perona., "Preattentive texture discrimination with early vision mechanisms," *Journal of the Optical Society of America A*, vol. 7, no. 5, pp. 923-932, 1990.

[37] J. Mao and A. K. Jain, "Texture classification and segmentation using multiresolution simultaneous autoregressive models," *Pattern Recognition*, vol. 25, no. 2, pp. 173-188, 1992.

[38] MatlabPyrTools-v1.4, http://www.cns.nyu.edu/~lcv/software.php

[39] R. Mehrotra and J.E. Gary, "Similar-shape retrieval in shape data management," *IEEE Comput.*, vol. 28, no. 9, pp. 57-62, 1995.

[40] T. Ojala, M. Pietikäinen and D. Harwood, "A Comparative Study of Texture Measures with Classification Based on Feature Distributions," *Pattern Recognition*, vol. 29, 51-59, 1996.
[41] T. Ojala, M. Pietikäinen, and T. Maenpaa, "Multiresolution Grey-Scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 24, pp. 971-987, 2002.
[42] A. V. Oppenheim and J. S. Lim, "The Importance of Phase in Signals," *Proceedings of the IEEE*, vol. 69, no. 5, pp. 529-541, 1981.
[43] J. M. Palomares, J. Gonzalez, E. Ros and A. Prieto, "General Logarithmic Image Processing Convolution", *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3602-3608, 2006.
[44] S. Panis, J. De Winter, J. Vandekerckhove, and J. Wagemans, "Identification of everyday objects on the basis of fragmented outline versions," *Perception*, vol. 37, pp. 271-289, 2008.
[45] A. Papoulis, *Probability, Random Variables, and Stochastic Processes (Second Edition)*, New York: McGraw-Hill, 1984.
[46] P.M. Pennefather, A. Chandna, I. Kovács, U. Polat, and A.M. Norcia, "Contour detection threshold: repeatability and learning with 'contour cards'," *Spatial Vision*., vol. 12, pp. 257-266, 1999.
[47] M.W. Pettet, S.P. McKee, and N.M. Grzywacz, "Constraints on long range interactions mediating contour detection," *Vision Research*, vol. 38, pp. 865-879, 1998.
[48] X. Qin and Y. Yang, "Basic Grey level aura matrices: theory and its application to texture synthesis," in *Proc. the Tenth IEEE International Conference on Computer Vision*, 2005, vol. 1, pp. 128-135.
[49] I. Sekita, T. Kurita, and N. Otsu, "Complex autoregressive model for shape recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 489-496, 1992.
[50] J. Serra, *Image analysis and mathematical morphology*, London: Academic Press, 1982.
[51] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Visual Recognition," *Proc. International Conference on Learning Representations*, 2015.
[52] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision*, NJ: Chapman & Hall, pp. 193-242, 1993.
[53] L. Spillmann and J. S. Werner, "Long-range interactions in visual perception," *Trends in Neurosciences*, vol. 19, pp. 428-434, 1996.
[54] M. Varma and A. Zisserman, "A Statistical Approach to Material Classification Using Image Patch Exemplars," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, pp. 2032-2047, 2009.
[55] J. Wagemans, J. De Winter, H. Op de Beeck, A. Ploeger, T. Beckers, and P. Vanroose, "Identification of everyday objects on the basis of silhouette and outline versions," *Perception*, vol. 37, pp. 207-244, 2008.
[56] L.Ying, "Phase unwrapping," *Wiley Encyclopedia of Biomedical Engineering*, vol. 6, pp. 1-11, 2006.
[57] D. Zhang and G .Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, pp. 1-19, 2004.

Xinghui Dong received his PhD degree from Heriot-Watt University, UK, in 2014. He is currently working as a Research Associate in the Centre for Imaging Sciences, The University of Manchester, UK. His research interests include automatic defect detection, image representation, texture analysis and visual perception.

Mike J. Chanter gained his PhD in 1994 studying the effect of illumination direction on image texture. Since then he has continued to study the many aspects of three-dimensional surface texture and its basic physical and perceptual dimensions, both from a computer vision point of view and also by borrowing rigorous methodology from psychophysics and experimental psychology. He founded and directs the Texture Lab at Heriot-Watt University and has helped organise many research events including chairing three international workshops on texture. He has supervised over twenty PhDs, published over 120 papers, and attracted considerable funding from EU and UK research directorates in this and related areas.